

(11)Publication number : 05-313923
(43)Date of publication of application : 26.11.1993

G06F 9/46
G06F 15/16

(71)Applicant : RICOH CO LTD

(72)Inventor : SASAKI JUN

(57)Abstract:

The block diagram illustrates the system architecture. On the left, three input modules labeled 'カメラA' (Camera A), 'カメラB' (Camera B), and 'カメラC' (Camera C) are connected via dashed lines to a central processing unit. This unit contains two main functional blocks: '画像取得処理部' (Image Acquisition Processing Unit) at the top and '画像認識処理部' (Image Recognition Processing Unit) at the bottom. To the right of this central unit is another module labeled '共有資源管理制御部' (Shared Resource Management Control Unit). This module contains a sub-unit labeled '共有資源情報記憶部' (Shared Resource Information Storage Unit). The entire system is enclosed within a larger boundary, with reference numerals 13 and 14 indicating specific components or sections.

DATE	TIME	FROM	TO	BY	REMARKS	REMARKS	REMARKS	REMARKS
1981	10:00	101	102	103	104	105	106	107

Therefore, a resources acquisition processing part 13 enables the tasks having the resources acquiring requests to periodically acquire the resources. Then, even the tasks issuing a request later can outrun the task that issues a request earlier according to the processing priority to acquire the resources.

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's

decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

Japanese Patent Application, Publication No. H05-313923

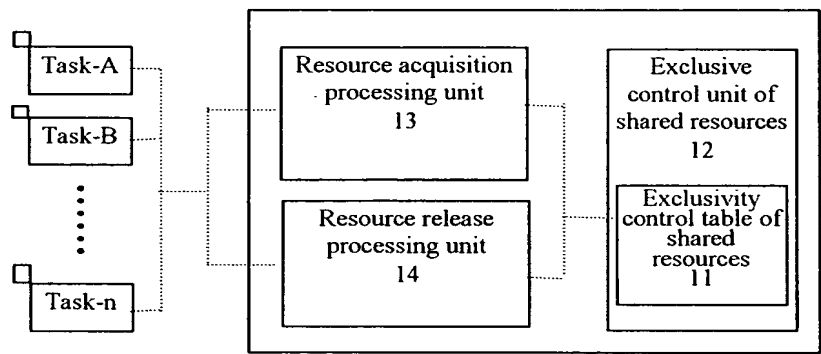
[0020] The resource acquisition processing unit 13, when a resource acquisition request is issued by any of the tasks, controls a resource acquisition process progressed in correspondence to this resource acquisition request by activating and causing the exclusive control unit 12 that is used for exerting exclusivity control of shared resources to enter relevant pieces of resource acquisition data into the exclusivity control table 11 that is used for exerting an exclusivity control of the shared resources. The resource release processing unit 14, when a resource release request is issued by any of the tasks, controls a resource release process progressed in corresponding to this resource release request by activating and causing the exclusive control unit 12 to enter relevant pieces of resource release data into the exclusivity control table 11. The exclusive control unit 12, here, when receiving an exclusivity control exertion request from either the resource acquisition processing unit 13 or the resource release processing unit 14, analyzes the received request and enters various pieces of data relevant to the received exclusivity control exertion request into the exclusivity control table 11 that is used for exerting the exclusivity control of the shared resources.

[0032] As indicated in Fig.2 (b), in the order same to the order of tasks issuing their requests, hence task-A, first, acquires resource- R_1 , and in reflection of this, respectively relevant pieces of data relating to the request issued by task-A are filled into entry fields from one titled as "priority order when in use" to one titled as "interval time between resource acquisition trials". And, then, task-C in the 2nd priority post and both task-B & task-D in the 3rd priority post are set to form the resource acquisition waiting queue reflecting this priority order.

[0033] When task-A completes the required process and issues a resource release request (of which the release mode is set as "normal"), task-C that is positioned at the top of the resource acquisition waiting queue of the tasks provided in Fig.2 (b) acquires resource- R_1 . This implies that, as indicated in Fig.2 (c), respectively relevant pieces of data relating to the request issued by task-C are filled into entry fields from one titled as "priority order when in use" to one titled as "interval time between resource acquisition trials". Then, task-B & task-D in the 3rd priority post are set to form the resource acquisition waiting queue reflecting this priority order.

Fig.1

(a)



(b)

Source name	Priority order of occupant	Priority order, latest position	Resource acquisition mode	Resource occupying period	Resource acquiring cycle time	Resource acquiring priority order	Queue of tasks waiting to acquire resources	Queue of tasks waiting for cyclic acquisition
(R)	(P)	(HP)	(M)	(OT)	(CT)	(RP)	(RQ)	(CQ)

Fig.2 (to continue to next page)

(a) Initialized state of system just after it is started up

Source name	Priority order of occupant	Priority order, latest position	Resource acquisition mode	Resource occupying period	Resource acquiring cycle time	Resource acquiring priority order	Queue of tasks waiting to acquire resources	Queue of tasks waiting for cyclic acquisition
R 1	0	0	0	0	0	1 RP1	None	None
						2 RP2	None	None
						3 RP3	None	None
						n RP n	None	None

Fig.2 (to be continued from previous page)

(b) State shown in Fig.2 (a) changes into following state when following 4 requests are issued by relevant tasks in the ascending order from (1) to (4)

- (1) Task A issues resource acquisition request (acquisition mode: ordinary, occupying time: 10, priority order: 2)
 (2) Task B issues resource acquisition request (acquisition mode: cyclic, occupying time: 20, cycle time: 50, priority order: 3)
 (3) Task C issues resource acquisition request (acquisition mode: ordinary, occupying time: 30, priority order: 2)
 (4) Task D issues resource acquisition request (acquisition mode: ordinary, occupying time: 40, priority order: 3)

* Task A acquires this resource

Source name	Priority order of occupant	Priority order, latest position	Resource acquisition mode	Resource occupying period	Resource acquiring cycle time	Resource acquiring priority order	Queue of tasks waiting to acquire resources	Queue of tasks waiting for cyclic acquisition
R 1	2	2	normal	10	0	1 RP1	None	None
						2 RP2	Task C	None
						3 RP3	Tasks B & D	None
						N RP n	None	None

(c) State shown in Fig.2 (b) changes into following state when Task A issues resource release request (release mode: normal)

* Task C acquires this resource

Source name	Priority order of occupant	Priority order, latest position	Resource acquisition mode	Resource occupying period	Resource acquiring cycle time	Resource acquiring priority order	Queue of tasks waiting to acquire resources	Queue of tasks waiting for cyclic acquisition
R 1	2	2	normal	30	0	1 RP1	None	None
						2 RP2	None	None
						3 RP3	Tasks B & D	None
						N RP n	None	None

(19)日本国特許庁(JP)

(12)公開特許公報(A)

(11)特許出願公開番号

特開平5-313923

(43)公開日 平成5年(1993)11月26日

(51)Int.Cl.⁴ 識別記号 庁内整理番号 F I 技術表示箇所
G 0 6 F 9/46 3 4 0 F 8120-5B
15/16 3 4 0 A 8840-5L

審査請求 未請求 請求項の数 5(全 20 頁)

(21)出願番号 特願平4-113842

(22)出願日 平成4年(1992)5月7日

(71)出願人 000006747

株式会社リコー

東京都大田区中馬込1丁目3番6号

(72)発明者 佐々木 潤

東京都大田区中馬込1丁目3番6号 株式

会社リコー内

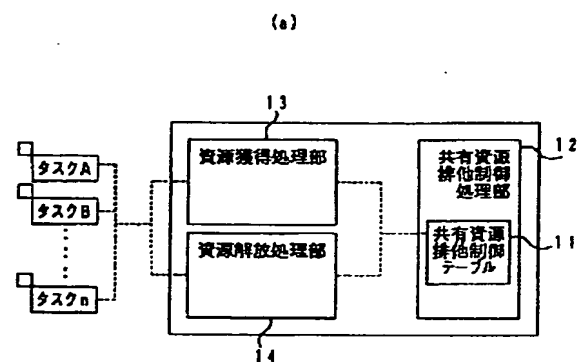
(74)代理人 弁理士 有我 軍一郎

(54)【発明の名称】 共有資源の排他制御装置

(57)【要約】

【目的】 タスクによって発行される資源獲得要求に優先順位を付加することにより、資源獲得の追い越し処理が可能となる共有資源の排他制御装置を提供することを目的としている。

【構成】 共有資源排他制御テーブル11は、複数のタスクA、B、…、nによって共有される資源に対して、各タスクから発行される資源獲得要求をその発行順に受け付けて待ち行列を作成する。ここで、前記資源獲得要求にはその発行時点でタスクにより優先順位が付加されており、この優先順位に従って共有資源排他制御処理部12は資源獲得要求の排他制御を実行する。



(b)

資源名	使用中優先順位	最新獲得優先順位	資源獲得モード	資源占有時間	資源解放時間	資源獲得待ち時間	資源獲得待ちタスクキュー	資源解放待ちタスクキュー
(R)	(P)	(HP)	(M)	(OT)	(CT)	(RP)	(RQ)	(CO)

【特許請求の範囲】

【請求項1】複数のタスクによって共有される資源に対して各タスクから発行される資源獲得要求をその発行順に受け付けて待ち行列を作成する共有資源排他制御テーブルと、

この待ち行列に従って1つの共有資源に対する複数タスクの同時アクセスを排除する共有資源排他制御処理部と、を備えた共有資源の排他制御装置において、前記資源獲得要求にその発行時点で優先順位を付加し、この優先順位に従って資源獲得要求の排他制御を実行することを特徴とする共有資源の排他制御装置。

【請求項2】請求項1記載の共有資源の排他制御装置において、

前記資源獲得要求の発行順に従って同一優先順位における資源獲得要求の排他制御を実行することを特徴とする共有資源の排他制御装置。

【請求項3】請求項1または2記載の共有資源の排他制御装置において、

前記資源獲得要求にその発行時点で資源獲得周期時間を付加し、この資源獲得周期時間に従ってその資源獲得要求を発行したタスクに周期的に資源を獲得させる資源獲得処理部を備えたことを特徴とする共有資源の排他制御装置。

【請求項4】請求項1または2記載の共有資源の排他制御装置において、

前記資源獲得要求を発行したタスクに強制的に資源を獲得させる資源獲得強制モードを設け、この強制モードでの資源獲得要求がいずれかのタスクから発行された場合に現在資源を獲得しているタスクを中断し前記共有資源排他制御テーブルの待ち行列の筆頭に該中断タスクを設定する資源獲得処理部を備えたことを特徴とする共有資源の排他制御装置。

【請求項5】請求項1または2記載の共有資源の排他制御装置において、

資源をタスクから強制的に解放する資源解放強制モードを設け、

この強制モードでの資源解放要求がいずれかのタスクから発行された場合に現在獲得されている資源を強制的に解放させて前記共有資源排他制御テーブルの待ち行列を初期化する資源解放処理部を備えたことを特徴とする共有資源の排他制御装置。

【発明の詳細な説明】**【0001】**

【産業上の利用分野】本発明は、例えば計算機システムやマルチプロセッサシステム等において、1つまたは複数の共有資源（例えばメモリ）に対して複数のタスク

（例えばプロセッサ）が同時にアクセスすることを排除する共有資源の排他制御装置に関し、特に複数のタスク間で共有するメモリエリア管理や、複数のタスクによる同一デバイスへのアクセスに好適な共有資源の排他制御

装置に関する。

【0002】

【従来の技術】従来、同一の共有資源に対して複数のタスクが同時にアクセスしないように、共有資源の排他制御装置が多用されている。従来のこの種の共有資源の排他制御装置としては、例えば特開平3-116261号公報、特開平3-141458号公報等に記載されたものがあり、排他制御により、複数のタスクで共有される資源（例えばメモリや各種デバイス等）に対して順次アクセスを実現している。

【0003】ここで、一般に知られている共有資源の排他制御は次のようなものである。ある特定のタスクが資源獲得要求を発行し、その資源が空いていれば（他のタスクにより獲得されていなければ）、資源獲得要求を発行したタスクに資源が渡され、空いていなければ資源獲得要求を発行したタスクは資源獲得待ちとなる。資源獲得待ちは、通常FIFO（First-in First-out）形式の待ち行列（キューイング）によって規定され、FIFO登録された順番に資源がタスクに渡される。このような待ち行列を規定したうえで、例えば現在資源を獲得しているタスクがその資源を解放すれば、待ち行列の先頭のタスクに資源が渡される。

【0004】通常、以上のような処理により資源が共有使用され、複数のタスクによって同時に資源にアクセスしないよう排他制御される。

【0005】

【発明が解決しようとする課題】しかしながら、このような従来の共有資源の排他制御装置にあっては、必ず資源獲得要求を発行した順に発行元タスクに資源が渡されるので、最後に獲得要求を発行したタスクは、待ち行列の最後尾にキューイングされ、このタスクに資源が割当てられるまで時間がかかる。また、最後尾にキューイングされたタスクに資源が割当てられるまでの時間は、現在資源を獲得しているタスクや前列にキューイングされているタスクの処理により左右され、その時間が変化する。

【0006】ここで、実際のシステムでは、種々のケースの発生が予想され、あるケースではFIFO順位によらない資源獲得が要求される場合がある。例えば、通常ならばタスクA、タスクBの順に資源が使用されるところでも、一旦エラーが発生したときにはタスクB、タスクAの順に資源を渡したい場合がある。ところが、従来の方式では後列にキューイングされたタスクが前列にキューイングされたタスクを追い越すことができず、このような処理の実現は不可能であった。

【0007】そこで、請求項1記載の発明は、タスクによって発行される資源獲得要求に優先順位を付加することにより、資源獲得の追い越し処理が可能となる共有資源の排他制御装置を提供することを目的としている。次に、資源獲得要求に優先順位を付加した場合、異なるタ

スクで同じ優先順位が付加される懸念がある。この場合、どちらを優先させるかの判断基準が必要になる。

【0008】そこで、請求項2記載の発明は、資源獲得要求の発行順に従って同一優先順位における資源獲得要求の排他制御を実行することにより、キューサータの時間短縮を可能とする共有資源の排他制御装置を提供することを目的としている。次に、従来の資源管理では周期的な資源割当は行われておらず、周期的にデータをサンプリングするようなときに不都合が発生していた。

【0009】そこで、請求項3記載の発明は、資源獲得要求に周期的な時間要素を付加することにより、特定のタスクに周期的に資源を割り当てることができる共有資源の排他制御装置を提供することを目的としている。次に、従来の制御方式では、何らかのエラーが発生し速やかに資源の初期化処理、再スタート処理等を行う必要があっても、資源が獲得できるまで時間がかかり速やかに処理できない事態が発生したり、エラーの発生によってあるタスクが資源を獲得したままダウンするような事態が発生しており、このような事態に対処して資源の強制獲得や強制解放が望まれる。

【0010】そこで、請求項4記載の発明は、資源獲得強制モードを設けることにより、速やかな復旧処理を可能とする共有資源の排他制御装置を提供することを目的としている。また、請求項5記載の発明は、資源解放強制モードを設けることにより、速やかな復旧処理を可能とする共有資源の排他制御装置を提供することを目的としている。

【0011】

【課題を解決するための手段】請求項1記載の発明は、上記目的を達成するために、複数のタスクによって共有される資源に対して各タスクから発行される資源獲得要求をその発行順に受け付けて待ち行列を作成する共有資源排他制御テーブルと、この待ち行列に従って1つの共有資源に対する複数タスクの同時アクセスを排除する共有資源排他制御処理部と、を備えた共有資源の排他制御装置において、前記資源獲得要求にその発行時点で優先順位を付加し、この優先順位に従って資源獲得要求の排他制御を実行することを特徴とする。

【0012】また、請求項2記載の発明は、上記目的を達成するために、請求項1記載の共有資源の排他制御装置において、前記資源獲得要求の発行順に従って同一優先順位における資源獲得要求の排他制御を実行することを特徴とする。また、請求項3記載の発明は、上記目的を達成するために、請求項1または2記載の共有資源の排他制御装置において、前記資源獲得要求にその発行時点で資源獲得周期時間を付加し、この資源獲得周期時間に従ってその資源獲得要求を発行したタスクに周期的に資源を獲得させる資源獲得処理部を備えたことを特徴とする。

【0013】また、請求項4記載の発明は、上記目的を

達成するために、請求項1または2記載の共有資源の排他制御装置において、前記資源獲得要求を発行したタスクに強制的に資源を獲得させる資源獲得強制モードを設け、この強制モードでの資源獲得要求がいずれかのタスクから発行された場合に現在資源を獲得しているタスクを中断し前記共有資源排他制御テーブルの待ち行列の筆頭に該中断タスクを設定する資源獲得処理部を備えたことを特徴とする。

【0014】また、請求項5記載の発明は、上記目的を達成するために、請求項1または2記載の共有資源の排他制御装置において、資源をタスクから強制的に解放する資源解放強制モードを設け、この強制モードでの資源解放要求がいずれかのタスクから発行された場合に現在獲得されている資源を強制的に解放させて前記共有資源排他制御テーブルの待ち行列を初期化する資源解放処理部を備えたことを特徴とする。

【0015】

【作用】上記構成を有する請求項1記載の発明においては、資源獲得要求にその発行時点で優先順位を付加し、この優先順位に従って共有資源排他制御処理部が資源獲得要求の排他制御を実行する。また、上記構成を有する請求項2記載の発明においては、資源獲得要求の発行順に従って共有資源排他制御処理部が同一優先順位における資源獲得要求の排他制御を実行する。

【0016】また、上記構成を有する請求項3記載の発明においては、資源獲得要求にその発行時点で資源獲得周期時間が付加されており、この資源獲得周期時間に従って資源獲得処理部がその資源獲得要求を発行したタスクに周期的に資源を獲得させる。また、上記構成を有する請求項4記載の発明においては、資源獲得要求を発行したタスクに強制的に資源を獲得させる資源獲得強制モードが設けられており、この強制モードでの資源獲得要求がいずれかのタスクから発行された場合に、資源獲得処理部は、現在資源を獲得しているタスクを中断し、共有資源排他制御テーブルの待ち行列の筆頭に該中断タスクを設定する。

【0017】また、上記構成を有する請求項5記載の発明においては、資源をタスクから強制的に解放する資源解放強制モードが設けられており、この強制モードでの資源解放要求がいずれかのタスクから発行された場合に、資源解放処理部は、現在獲得されている資源を強制的に解放させて、共有資源排他制御テーブルの待ち行列を初期化する。

【0018】

【実施例】以下、本発明を実施例に基づいて説明する。図1は請求項1～5いずれかに記載された発明の一実施例に係る共有資源の排他制御装置を示す図であり、同図(a)はそのブロック構成図、同図(b)は同図(a)に示す共有資源排他制御テーブルの構成図である。

【0019】まず、構成を説明する。図1(a)におい

て、タスクA、B、…、nは、例えばメモリ等の共有資源を使用するために、共有資源の獲得／解放を行う例えばプロセッサ群を示している。共有資源排他制御テーブル11は、複数のタスクA、B、…、nによって共有される資源に対して、各タスクから発行される資源獲得要求をその発行順に受け付けて待ち行列を作成する。共有資源排他制御処理部12は、この待ち行列に従って1つの共有資源に対する複数タスクの同時アクセスを排除する。なお、本実施例では共有資源排他制御処理部12にテーブル11を内蔵しているが、処理部12とテーブル11を別構成としても構わない。

【0020】資源獲得処理部13は、各タスクから資源獲得要求が発行された時、共有資源排他制御処理部12を用いて共有資源排他制御テーブル11に資源獲得情報を設定し、資源獲得を制御する。資源解放処理部14は、各タスクからの資源解放要求が発行された時、共有資源排他制御処理部12を用いて共有資源排他制御テーブル11に資源解放情報を設定し、資源解放を制御する。なお、共有資源排他制御処理部12は、資源獲得処理部13および資源解放処理部14からの共有資源排他制御要求に対してその要求を解析し、共有資源排他制御テーブル11に各種情報を設定する。

【0021】ここで、前記資源獲得要求にはその発行時点でタスクにより優先順位が付加されており、この優先順位に従って共有資源排他制御処理部12は資源獲得要求の排他制御を実行する。また、前記資源獲得要求の発行順に従って共有資源排他制御処理部12は同一優先順位における資源獲得要求の排他制御を実行する。

【0022】次に、前記資源獲得要求にはその発行時点でタスクにより資源獲得周期時間を付加することができ、ここで付加された資源獲得周期時間に従って資源獲得処理部13はその資源獲得要求を発行したタスクに周期的に資源を獲得させる。また、前記資源獲得要求を発行したタスクに強制的に資源を獲得させる資源獲得強制モードが設けられており、この強制モードでの資源獲得要求がいずれかのタスクから発行された場合に、資源獲得処理部13は現在資源を獲得しているタスクを中断し、前記共有資源排他制御テーブル11の待ち行列の筆頭に該中断タスクを設定する。

【0023】また、資源をタスクから強制的に解放する資源解放強制モードが設けられており、この強制モードでの資源解放要求がいずれかのタスクから発行された場合に、資源解放処理部14は現在獲得されている資源を強制的に解放させて、前記共有資源排他制御テーブル11の待ち行列を初期化する。共有資源排他制御テーブル11の管理項目が図1(b)に示される。このテーブル11は、資源名をキーとして各資源毎に項目データを管理する。図示のように、資源名に対応させて、使用中優先順位、最新高優先順位、資源獲得モード、資源占有時間、資源獲得周期時間、資源獲得優先順位、資源獲得待

ちタスクキュー、周期獲得待ちタスクキュー、の各管理項目が設定されており、各々次に説明する目的のために使用される。

【0024】資源名：R

複数資源の管理を行う時の識別子として用いる。このため、共有資源排他制御テーブル11は、資源名毎に資源数に応じて複数作成される。

使用中優先順位：P

現在資源を獲得しているタスクの資源獲得優先順位を格納し、後から要求される資源獲得優先順位との比較のために使用する。また、この使用中優先順位が設定されているか否かによって、資源の使用（＝使用中優先順位≠初期値）／未使用（初期値＝例えば「0」）を判別する。

【0025】最新高優先順位：HP

資源が使用中の時に、他のタスクから要求された資源獲得優先順位と使用中タスクの優先順位を比較し、より高い方の優先順位を格納する。すなわち、使用中の資源が解放された時、次に資源を割当てるタスクの優先順位を示す。

資源獲得モード：M

通常獲得モード、周期獲得モード、強制獲得モード等の識別に用いる。ここで、強制モードでの資源獲得要求がいずれかのタスクから発行された場合に、資源獲得処理部13は現在資源を獲得しているタスクを中断し、前記共有資源排他制御テーブル11の待ち行列の筆頭に該中断タスクを設定する。そして、前記資源獲得要求を発行したタスクに強制的に資源を獲得させる。また、強制モードでの資源解放要求がいずれかのタスクから発行された場合に、資源解放処理部14は現在獲得されている資源をタスクから強制的に解放させて、前記共有資源排他制御テーブル11の待ち行列を初期化する。

【0026】資源占有時間：OT

資源獲得要求を発行したタスクの最大資源占有時間を設定する。いかなるタスクもこの占有時間を越えて資源を保持することはできない。この占有時間を越えたときは、このタスクによって使用されている資源を解放して、次の待ち行列（キュー）に繋がれているタスクに資源を割当てる。なお、占有時間を越えたタスクは要求を発行した時点のモードで新たに待ち行列にリンクされる。

【0027】資源獲得周期時間：CT

資源の周期獲得を行う時の周期時間を設定する。すなわち、この資源獲得周期時間に従って、資源獲得処理部13はその資源獲得要求を発行したタスクに周期的に資源を獲得させる。

資源獲得優先順位：RP

優先順位は1～mの番号で示され、番号が小さいほど優先順位が高い。ただし、RP＝1の最高位優先順位番号は強制獲得モードで使用するものとし、通常は2～mの

番号が使用される。なお、 $RP=1$ の場合、必然的に前記資源獲得モード(M)は強制モードの識別子が設定される。前記共有資源排他制御処理部12は、この優先順位に従って資源獲得要求の排他制御を実行する。

【0028】資源獲得待ちタスクキュー：RQ

同一の資源獲得優先順位を有する他のタスクが既に資源を獲得している時、通常獲得モード要求を発行したタスクがこのキューにリンク(エンキュー)される。共有資源排他制御処理部12は、本資源獲得待ちタスクキューに従って、同一優先順位における資源獲得要求の発行順に排他制御を実行する。

【0029】周期獲得待ちタスクキュー：CQ

同一の資源獲得優先順位を有する他のタスクが既に資源を獲得している時、周期獲得モード要求(CT)を発行したタスクがこのキューにリンク(エンキュー)される。そして、資源解放後に資源獲得処理部13によってCQのタスクが資源を獲得し、結果的にCQのタスクが周期的に資源を獲得する。

【0030】次に、作用を説明する。まず、図2～図4を参照しながら図1に示した共有資源排他制御テーブルの使用状況を示す動作例を説明する。なお、図2～図4は連続した動作を示す。図2(a)はシステムスタートアップ時の初期化状態を示し、資源名R1以外は「0」または「無し」の初期値がセットされている。

【0031】ここで、例えばA～Dの各タスクから次の(1)～(4)の順で各要求が発行されると、(1)タスクAが資源獲得要求(獲得モード：通常、占有時間：10、優先順位：2)を発行、(2)タスクBが資源獲得要求(獲得モード：周期、占有時間：20、優先順位：3、周期時間：50)を発行、(3)タスクCが資源獲得要求(獲得モード：通常、占有時間：30、優先順位：2)を発行、(4)タスクDが資源獲得要求(獲得モード：通常、占有時間：40、優先順位：3)を発行。

【0032】図2(b)に示すように、要求の発行順にまずタスクAが資源R1を獲得し、使用中優先順位～資源獲得周期時間までの項目にそれぞれタスクAの要求データがセットされる。次いで、資源獲得優先順位に従って、優先順位：2にタスクCが、また優先順位：3にタスクBおよびDが、それぞれ資源獲得待ちタスクキューとしてセットされる。

【0033】その後、タスクAが処理を終了し、資源解放要求(解放モード：通常)を発行すると、図2(b)の資源獲得待ちタスクキューの先頭にキューイングされたタスクCが資源R1を獲得し、図2(c)に示すように、使用中優先順位～資源獲得周期時間までの項目にそれぞれタスクCの要求データがセットされる。次いで、資源獲得優先順位に従って、優先順位：3のタスクBおよびDが、そのまま資源獲得待ちタスクキューとしてセットされる。

【0034】その後、タスクCが処理を終了し、資源解放要求(解放モード：通常)を発行すると、図2(c)の資源獲得待ちタスクキューの先頭にキューイングされたタスクBが資源R1を獲得し、図3(a)に示すように、使用中優先順位～資源獲得周期時間までの項目にそれぞれタスクBの要求データがセットされる。次いで、資源獲得優先順位に従って、優先順位：3のタスクDが資源獲得待ちタスクキューとしてセットされる。なお、タスクDに先行してタスクBが資源R1を獲得した理由は、その要求発行順による。

【0035】その後、タスクBが処理を終了し、資源解放要求(解放モード：通常)を発行すると、図3(a)の資源獲得待ちタスクキューの先頭にキューイングされたタスクDが資源R1を獲得し、図3(b)に示すように、使用中優先順位～資源獲得周期時間までの項目にそれぞれタスクDの要求データがセットされる。この時点で資源獲得待ちタスクキューとしてセットされるタスクはなくなる。しかし、タスクBは資源獲得周期時間：50がセットされているので、優先順位：3の周期獲得待ちタスクキューにセットされ、また資源獲得周期時間：50はそのままとされる。

【0036】その後、タスクDの処理中(資源占有時間：40内)に、タスクBの資源獲得周期時間：50がオーバーすると、図3(c)に示すように、優先順位：3においてタスクBが周期獲得待ちタスクキューから資源獲得待ちタスクキューに変更される。なお、この時点ではタスクDが資源R1を保持したままとなる。本実施例は周期時間がセットされているタスクBが資源獲得待ちタスクキューにセットされた時点で、タスクBとタスクDが入れ替わり、タスクBの周期的資源獲得を可能とするものであるが、この周期的資源獲得よりも優先するのが強制獲得モードである。

【0037】すなわち、図3(c)の状態ではタスクAが強制モードによる資源獲得要求(占有時間：60)を発行すると、図4(a)に示すように、タスクAが資源R1を獲得し、タスクDが処理を中断してタスクBの前列にキューイングされる。一方、図3(c)の状態では順当にタスクBが資源を獲得した状態が図4(b)に示される。すなわち、タスクBとタスクDが入れ替わり、周期的にタスクBが資源R1を獲得する。

【0038】なお、強制モードでの資源解放要求がいずれかのタスクから発行された場合は、共有資源排他制御テーブル11がどのような状態にあっても、図2(a)に示した状態に初期化される。次に、図5～図12を参照しながら図1に示す排他制御装置の各処理を説明する。なお、以下の説明では、本発明の特徴である共有資源排他制御処理部12(共有資源排他制御テーブル11)の動作を中心に説明する。

【0039】図5は図1に示す排他制御装置のメイン処理を示すフローチャートであり、同図(a)はタスクが

らの要求に基づいて実処理を選択するフロー、同図

(b) は占有時間や周期時間の監視対象時間がタイムオーバーした時に起動される実処理の選択フローである。図5(a)において、タスクから獲得または解放の資源制御要求が発行されるのを待機し(ステップS1)、要求があると、共有資源排他制御処理部12は、タスクによって要求された資源名をキーとして共有資源排他制御テーブル11をサーチする(ステップS2)。テーブルを特定すると、タスクからの要求内容を解析し(ステップS3)、優先順位により資源を獲得する通常獲得処理(ステップS4)、周期時間により資源を獲得する周期獲得処理(ステップS5)、強制モードにより資源を獲得する強制獲得処理(ステップS6)、現在資源を獲得しているタスクからの要求により資源を解放する通常解放処理(ステップS7)、または強制モードにより資源を解放する強制解放処理(ステップS8)、いずれかを選択する。

【0040】図5(a)に示す要求発行順の処理選択と共に、図5(b)に示す時間監視による処理選択が行われる。すなわち、監視時間のタイムオーバーを待機し(ステップT1)、タイムオーバーになると、その監視時間の種別を判断する(ステップT2)。ここで、監視時間が占有時間である場合は、現在タスクにより獲得されている資源を時間切れにより解放する占有時間オーバー処理(ステップT3)、を実行する。また、監視時間が周期時間である場合は、現在資源を獲得しているタスクから周期要求タスクに引き渡す周期時間オーバー処理(ステップT4)、を実行する。

【0041】図6は図5(a)のステップS4に示す通常獲得処理の動作フローである。通常獲得処理は、あるタスクが通常獲得モードで資源獲得要求を発行した時に起動される。タスクからの資源獲得要求は、資源獲得処理部13で認識され、共有資源排他制御処理部12で解析される。共有資源排他制御処理部12では、要求資源名と同一の共有資源排他制御テーブル11を捜し出し、使用中優先順位を調べる(ステップP1)。ここで、使用中優先順位P=初期値すなわち資源が未使用ならば、共有資源排他制御テーブル11の使用中優先順位Pおよび最新高優先順位HP両方に、タスクから要求された資源獲得優先順位RPを設定し(ステップP2)、資源獲得モードおよび資源占有時間には各々の要求値を設定して(ステップP3、P4)、要求発行元のタスクに資源を割当てる。

【0042】一方、ステップP1の判断で使用中優先順位P≠初期値すなわち資源が使用中であれば、要求された資源獲得優先順位RPと使用中優先順位Pを比較し(ステップP5)、より高い方の優先順位を最新高優先順位HPに設定し(ステップP6、P7)、要求された資源獲得優先順位RPが示す資源獲得待ちタスクキューRQの最後に要求発行元のタスクをエンキューして資源

獲得待ち状態とする(ステップP8)。

【0043】ここで、要求発行元のタスクがエンキューされる待ち行列は、その資源獲得優先順位RPの値により規定され、仮に低い優先順位RPのタスクが先に発行されていても、この低優先順位のタスクを飛び越した位置にエンキューされる。このように、請求項1記載の実施例においては、優先順位により資源が獲得されるので、後から要求を発行したタスクでも処理の優先度によって先に要求を発行したタスクを追い越して資源を獲得することが可能となる。

【0044】また、同一優先順位のタスクは要求の発行順に待ち行列にエンキューされる。このように、請求項2記載の実施例においては、優先順位毎のプライオリティキューイングにより、資源獲得/解放等の一連のテーブルサーチ処理の高速化が可能となる。図7は図5

(a)のステップS5に示す周期獲得処理の動作フローである。

【0045】周期獲得処理は、あるタスクが周期獲得モードで資源獲得要求を発行した時に起動される。図6の処理と同様に、タスクからの資源獲得要求は、資源獲得処理部13で認識され、共有資源排他制御処理部12で解析される。共有資源排他制御処理部12では、要求資源名と同一の共有資源排他制御テーブル11を捜し出し、使用中優先順位を調べる(ステップQ1)。ここで、使用中優先順位P=初期値すなわち資源が未使用ならば、共有資源排他制御テーブル11の使用中優先順位Pおよび最新高優先順位HP両方に、タスクから要求された資源獲得優先順位RPを設定し(ステップQ2)、資源獲得モードおよび資源占有時間には各々の要求値を設定する(ステップQ3、Q4)。さらに、資源獲得周期時間CTに周期時間を設定して(ステップQ5)、要求発行元のタスクに資源を割当てる。

【0046】一方、ステップQ1の判断で使用中優先順位P≠初期値すなわち資源が使用中であれば、要求された資源獲得優先順位RPと使用中優先順位Pを比較し(ステップQ6)、より高い方の優先順位を最新高優先順位HPに設定し(ステップQ7、Q8)、要求された資源獲得優先順位RPが示す資源獲得待ちタスクキューRQの最後に要求発行元のタスクをエンキューして資源獲得待ち状態とする(ステップQ9)。

【0047】このように、請求項3記載の実施例においては、資源獲得周期時間CTによる資源管理により、タスクは優先順位に従って周期的に資源を獲得することができ、周期的に資源を獲得して処理するデータサンプリング等が可能となる。図8は図5(a)のステップS6に示す強制獲得処理の動作フローである。強制獲得処理は、あるタスクが強制獲得モードで資源獲得要求を発行した時に起動される。図6の処理と同様に、タスクからの資源獲得要求は、資源獲得処理部13で認識され、共有資源排他制御処理部12で解析される。共有資

源排他制御処理部12では、要求資源名と同一の共有資源排他制御テーブル11を捜し出し、使用中優先順位を調べる(ステップR1)。ここで、使用中優先順位P=初期値すなわち資源が未使用ならば、共有資源排他制御テーブル11の使用中優先順位Pおよび最新高優先順位HP両方に、タスクから要求された資源獲得優先順位RP=1(最高優先順位)を設定し(ステップR2)、資源獲得モードおよび資源占有時間には各々の要求値を設定して(ステップR3、R4)、要求発行元のタスクに資源を割当てる。

【0048】一方、ステップR1の判断で使用中優先順位P≠初期値すなわち資源が使用中であれば、現在資源を獲得しているタスクをその使用中優先順位Pにて示される該当優先順位の資源獲得待ちタスクキューの先頭へエンキューして資源獲得待ち状態とする(ステップR5)。次に、資源排他制御テーブル11の使用中優先順位に「1」を設定し(ステップR6)、資源獲得モードおよび資源占有時間には各々の要求値を設定して要求発行元のタスクに資源を割当てる(ステップR7、R8)。

【0049】このように、請求項4記載の実施例においては、資源獲得強制モードによる資源獲得により、資源獲得の待ち時間無しに資源を獲得することができ、資源に対する迅速な処理が可能となる。図9は図5(a)のステップS7に示す通常解放処理の動作フローである。通常解放処理は、あるタスクが通常解放モードで資源解放要求を発行した時に起動される。タスクからの資源解放要求は、資源解放処理部14で認識され、共有資源排他制御処理部12で解析される。共有資源排他制御処理部12では、要求資源名と同一の資源排他制御テーブル11を捜し出す。次に、最新高優先順位HPが示す資源獲得優先順位から優先順位の低い方へ資源獲得待ちタスクキューをサーチし(ステップU1)、サーチした待ち行列に獲得待ちしているタスクがあるか否かを判断する(ステップU2)。

【0050】ここで、獲得待ちしているタスクがある場合は、その待ちキューから先頭のタスクを抽出し(ステップU3)、この抽出したタスクに資源を割当てる(ステップU4)。すなわち、資源を割り当てられた抽出タスクをサーチした場所の資源獲得優先順位RPを、使用中優先順位Pと最新高優先順位HP両方に設定する。次に、抽出タスクが要求を発行した時点での要求モードを判定し(ステップU5)、通常獲得モードであれば、資源獲得モードMに「通常」を(ステップU6)、また資源占有時間OTに要求値を設定する(ステップU7)。

【0051】一方、ステップU5の判定で周期獲得モードであれば、資源獲得モードMに「周期」を(ステップU8)、資源占有時間OTに要求値を(ステップU9)、また資源獲得周期時間CTにその周期時間を設定する(ステップU10)。ただし、ステップU2の判断

で待ちキューに繋がれているタスクが無い時は、使用中優先順位Pを初期化して(ステップU11)、処理を終了する。

【0052】図10は図5(a)のステップS8に示す強制解放処理の動作フローである。強制解放処理は、あるタスクが強制解放モードで資源解放要求を発行した時に起動される。図9の処理と同様に、タスクからの資源解放要求は、資源解放処理部14、共有資源排他制御処理部12で認識、解析される。共有資源排他制御処理部12では、要求資源名の資源排他制御テーブル11を捜し出し(ステップV1)、その資源排他制御テーブル11にタスクがリンクされているか否かを判断する(ステップV2)。ここで、テーブル11にタスクがリンクされている場合は、リンクされている全てのタスクに対し資源が強制解放されたことを通知し、この資源排他制御テーブル11を初期化する(ステップV3)。

【0053】一方、ステップV2の判断で、テーブル11にタスクがリンクされていない場合は、即座に資源排他制御テーブル11を初期化する(ステップV4)。このように、請求項5記載の実施例においては、資源解放強制モードによる資源解放により、あるタスクが資源を獲得したままダウンしたような場合のエラーリカバリーが可能となる。

【0054】図11は図5(b)のステップT3に示す占有時間オーバー処理の動作フローである。占有時間オーバー処理は、資源を獲得しているタスクの資源占有時間OTがオーバーした時に起動される。資源占有時間OTをオーバーしたタスクには、資源占有時間がオーバーしたことを通知し(ステップW1)、このタスクを資源獲得要求を発行した時点のモードで待ちキューの最後にエンキューする(ステップW2)。

【0055】次に、最新高優先順位HPが示す資源獲得優先順位から優先順位の低い方へ資源獲得待ちまたは周期獲得待ちのタスクに繋がれているタスクをサーチし(ステップW3)、待ちキューから先頭のタスクを抽出して(ステップW4)、このタスクに資源を割当てる(ステップW5)。すなわち、資源を割り当てられた抽出タスクをサーチした場所の資源獲得優先順位RPを、使用中優先順位Pと最新高優先順位HP両方に設定する。

【0056】次に、抽出タスクが要求を発行した時点での要求モードを判定し(ステップW6)、通常獲得モードであれば、資源獲得モードMに「通常」を(ステップW7)、また資源占有時間OTに要求値を設定する(ステップW8)。一方、ステップW6の判定で周期獲得モードであれば、資源獲得モードMに「周期」を(ステップW9)、資源占有時間OTに要求値を(ステップW10)、また資源獲得周期時間CTにその周期時間を設定する(ステップW11)。

【0057】図12は図5(b)のステップT4に示す

周期時間オーバー処理の動作フローである。周期時間オーバー処理は、資源を周期獲得モードで使用するタスクの資源獲得周期時間がオーバーした時に起動される。まず、該当する排他制御テーブル11の使用優先順位Pを調べ(ステップX1)、P=初期値すなわち資源が未使用ならば、共有資源排他制御テーブル11の使用優先順位Pおよび最新高優先順位HP両方に、タスクが要求を発行した時点での資源獲得優先順位RPを設定し(ステップX2)、資源獲得モードおよび資源占有時間には各々の要求値を設定する(ステップX3、X4)。さらに、資源獲得周期時間CTに要求値を設定して(ステップX5)、周期獲得要求の発行元であるタスクに資源を割当てる。

【0058】一方、ステップX1の判断で使用中優先順位P≠初期値すなわち資源が使用中であれば、周期獲得待ちタスクキューCQに繋がれている該タスクを抽出し(ステップX6)、資源獲得待ちタスクキューRQにエンキューして(ステップX7)、資源が割当てられるのを待つ。また、このとき抽出したタスクを周期獲得待ちタスクキューCQの最後にエンキューして次の周期起動待ちとする。

【0059】

【発明の効果】以上説明したように、請求項1記載の発明に係る共有資源の排他制御装置によれば、資源獲得要求にその発行時点で優先順位を付加し、この優先順位に従って共有資源排他制御処理部が資源獲得要求の排他制御を実行するので、後から要求を発行したタスクでも処理の優先度によって先に要求を発行したタスクを追いついて資源を獲得することが可能となる。このため、共有資源の排他制御における資源の獲得/解放が迅速かつ確実に行われるようになり、従来システムで発生していた資源獲得待ちのための応答遅延やシステムダウンを最小限にとどめ共有資源の効率の良い逐次制御が可能となる。

【0060】また、請求項2記載の発明に係る共有資源の排他制御装置によれば、資源獲得要求の発行順に従って共有資源排他制御処理部が同一優先順位における資源獲得要求の排他制御を実行するので、資源獲得/解放等の一連のテーブルサーチ処理の高速化が可能となる。また、請求項3記載の発明に係る共有資源の排他制御装置によれば、資源獲得要求にその発行時点で資源獲得周期時間が付加されており、この資源獲得周期時間に従って資源獲得処理部がその資源獲得要求を発行したタスクに周期的に資源を獲得させるので、タスクは優先順位に従って周期的に資源を獲得することができ、周期的に資源を獲得して処理するデータサンプリング等が可能となる。

【0061】また、請求項4記載の発明に係る共有資源の排他制御装置によれば、資源獲得要求を発行したタスクに強制的に資源を獲得させる資源獲得強制モードが設

けられており、この強制モードでの資源獲得要求がいずれかのタスクから発行された場合に、資源獲得処理部は、現在資源を獲得しているタスクを中断し、共有資源排他制御テーブルの待ち行列の筆頭に該中断タスクを設定するので、資源獲得の待ち時間無しに資源を獲得することができ、資源に対する迅速な処理が可能となる。

【0062】また、請求項5記載の発明に係る共有資源の排他制御装置によれば、資源をタスクから強制的に解放する資源解放強制モードが設けられており、この強制モードでの資源解放要求がいずれかのタスクから発行された場合に、資源解放処理部は、現在獲得されている資源を強制的に解放させて、共有資源排他制御テーブルの待ち行列を初期化するので、あるタスクが資源を獲得したままダウンしたような場合のエラーリカバリーが可能となる。

【図面の簡単な説明】

【図1】請求項1～5いずれかに記載された発明の一実施例に係る共有資源の排他制御装置を示す図であり、同図(a)はそのブロック構成図、同図(b)は同図(a)に示す共有資源排他制御テーブルの構成図である。

【図2】図1に示した共有資源排他制御テーブルの使用状況を示す管理項目の変遷図であり、同図(a)はシステムスタートアップ時の初期化状態、同図(b)は初期化状態から発行順にタスクAが資源を獲得した状態、同図(c)はタスクAから優先順位によりタスクCが資源を獲得した状態を示す。

【図3】図2に続く管理項目の変遷図であり、同図(a)はタスクCから優先順位および発行順によりタスクBが資源を獲得した状態、同図(b)はタスクBから発行順によりタスクDが資源を獲得した状態、同図(c)は周期時間オーバーによりタスクBのキューが変更された状態を示す。

【図4】図3に続く管理項目の変遷図であり、同図(a)は図3(c)におけるタスクAによる資源の強制獲得状態を示し、同図(b)は図3(c)におけるタスクBの資源の周期的獲得状態を示す。

【図5】図1に示す排他制御装置のメイン処理を示すフローチャートであり、同図(a)はタスクからの要求に基づいて実処理を選択するフロー、同図(b)は占有時間や周期時間の監視対象時間がタイムオーバーした時に起動される実処理の選択フローである。

【図6】図5(a)のステップS4に示す通常獲得処理の動作フローである。

【図7】図5(a)のステップS5に示す周期獲得処理の動作フローである。

【図8】図5(a)のステップS6に示す強制獲得処理の動作フローである。

【図9】図5(a)のステップS7に示す通常解放処理の動作フローである。

【図10】図5(a)のステップS8に示す強制解放処理の動作フローである。

【図11】図5(b)のステップT3に示す占有時間オーバー処理の動作フローである。

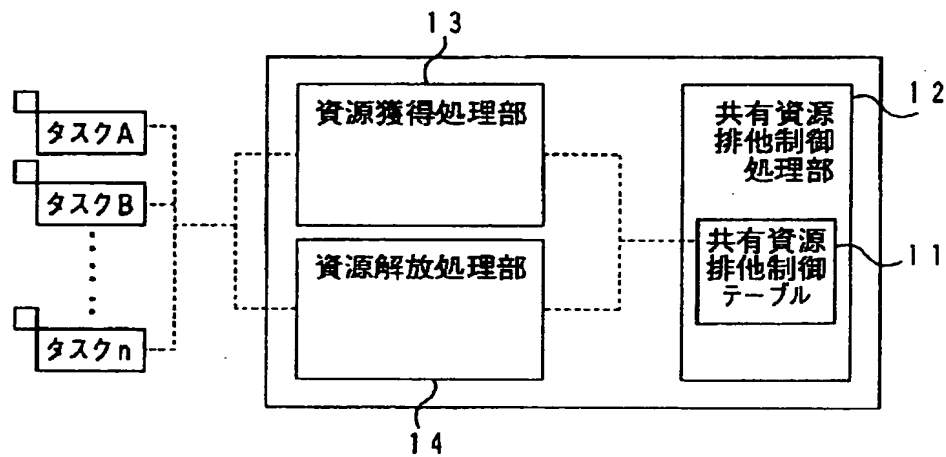
【図12】図5(b)のステップT4に示す周期時間オーバー処理の動作フローである。

【符号の説明】

- 11 共有資源排他制御テーブル
- 12 共有資源排他制御処理部
- 13 資源獲得処理部
- 14 資源解放処理部

【図1】

(a)



(b)

資源名	使用中 優先順位	最新高 優先順位	資源獲得 モード	資源 占有時間	資源獲得 周期時間	資源獲得 優先順位	資源獲得 待ち タスク キュー	周期獲得 待ち タスク キュー
(R)	(P)	(HP)	(M)	(OT)	(CT)	(RP)	(RQ)	(CQ)

【図2】

(a) システムスタートアップ時の初期化状態

資源名	使用中優先順位	最新高優先順位	資源獲得モード	資源占有時間	資源獲得周期時間	資源獲得優先順位	資源獲得待ち タスクキュー	周期獲得待ち タスクキュー
						1 RP1	無し	無し
R1	0	0	0	0	0	2 RP2	無し	無し
						3 RP3	無し	無し
						:		
						n R P n	無し	無し

(b)

(a) の状態で、各タスクが次の(1)～(4)の順で各要求を発行した時の状態

(1) タスクAが資源獲得要求(獲得モード:通常、占有時間:10、優先順位:2)発行

(2) タスクBが資源獲得要求(獲得モード:周期、占有時間:20、周期時間:50、優先順位:3)発行

(3) タスクCが資源獲得要求(獲得モード:通常、占有時間:30、優先順位:2)発行

(4) タスクDが資源獲得要求(獲得モード:通常、占有時間:40、優先順位:3)発行

※ タスクAが資源を獲得

資源名	使用中優先順位	最新高優先順位	資源獲得モード	資源占有時間	資源獲得周期時間	資源獲得優先順位	資源獲得待ち タスクキュー	周期獲得待ち タスクキュー
						1 RP1	無し	無し
R1	2	2	通常	10	0	2 RP2	タスクC	無し
						3 RP3	タスクB、タスクD	無し
						:		
						n R P n	無し	無し

(c)

(b) の状態で、タスクAが資源解放要求(解放モード:通常)を発行した時の状態

※ タスクCが資源を獲得

資源名	使用中優先順位	最新高優先順位	資源獲得モード	資源占有時間	資源獲得周期時間	資源獲得優先順位	資源獲得待ち タスクキュー	周期獲得待ち タスクキュー
						1 RP1	無し	無し
R1	2	2	通常	30	0	2 RP2	無し	無し
						3 RP3	タスクB、タスクD	無し
						:		
						n R P n	無し	無し

【図3】

(a)

図2(c)の状態で、タスクCが資源解放要求(解放モード:通常)を発行した時の状態
 ※ タスクBが資源を獲得

資源名	使用中優先順位	最新高優先順位	資源獲得モード	資源占有時間	資源獲得周期時間	資源獲得優先順位	資源獲得待ちタスクキュー	周期獲得待ちタスクキュー
						1	RP1	無し
						2	RP2	無し
						3	RP3	タスクD
						:		
R1	3	3	周期	20	50	n	RPn	無し

(b)

(a)の状態で、タスクBが資源解放要求(解放モード:通常)を発行した時の状態
 ※ タスクDが資源を獲得

資源名	使用中優先順位	最新高優先順位	資源獲得モード	資源占有時間	資源獲得周期時間	資源獲得優先順位	資源獲得待ちタスクキュー	周期獲得待ちタスクキュー
						1	RP1	無し
						2	RP2	無し
						3	RP3	無し
						:		タスクB
R1	3	3	通常	40	50	n	RPn	無し

(c)

(b)の状態で、資源獲得周期時間がオーバーした時の状態
 ※ タスクDが資源を保持

資源名	使用中優先順位	最新高優先順位	資源獲得モード	資源占有時間	資源獲得周期時間	資源獲得優先順位	資源獲得待ちタスクキュー	周期獲得待ちタスクキュー
						1	RP1	無し
						2	RP2	無し
						3	RP3	タスクB
						:		
R1	3	3	通常	40	50	n	RPn	無し

【図 4】

(a)

図 3 (c) の状態で、タスク A が資源獲得要求 (獲得モード: 強制、占有時間: 60) を発行した時の状態

※ タスク A が資源を獲得.....タスク A が資源を解放すれば図 3 (c) の状態となる

資源名	使用中優先順位	最新高優先順位	資源獲得モード	資源占有時間	資源獲得周期時間	資源獲得優先順位	資源獲得待ちタスクキュー	周期獲得待ちタスクキュー
						1	RP1	無し
R1	1	3	強制	60	50	2	RP2	無し
						3	RP3	タスク D、タスク B
						:		
						n	RPn	無し

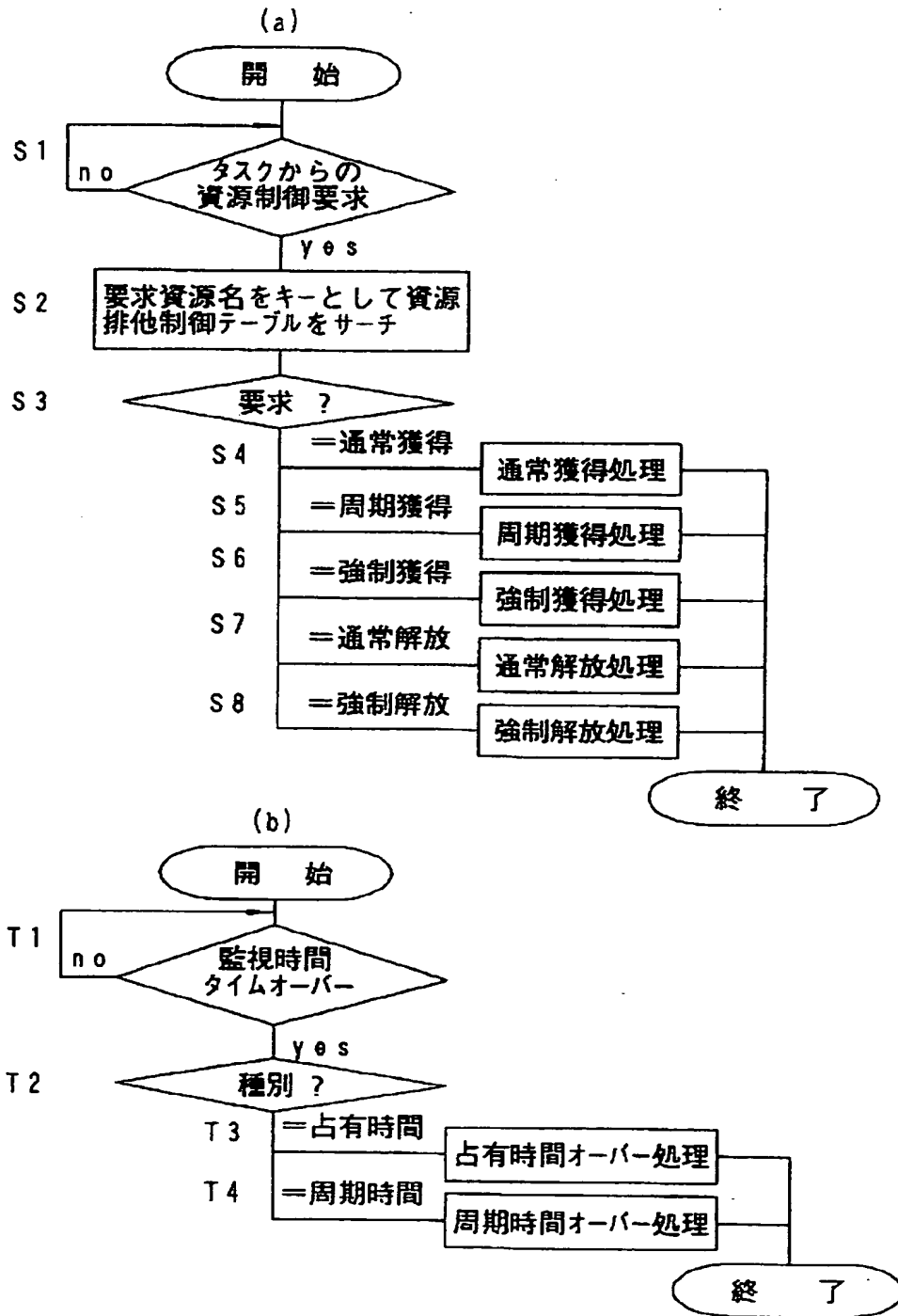
(b)

図 3 (c) の状態で、タスク D の資源占有時間がオーバーした時の状態

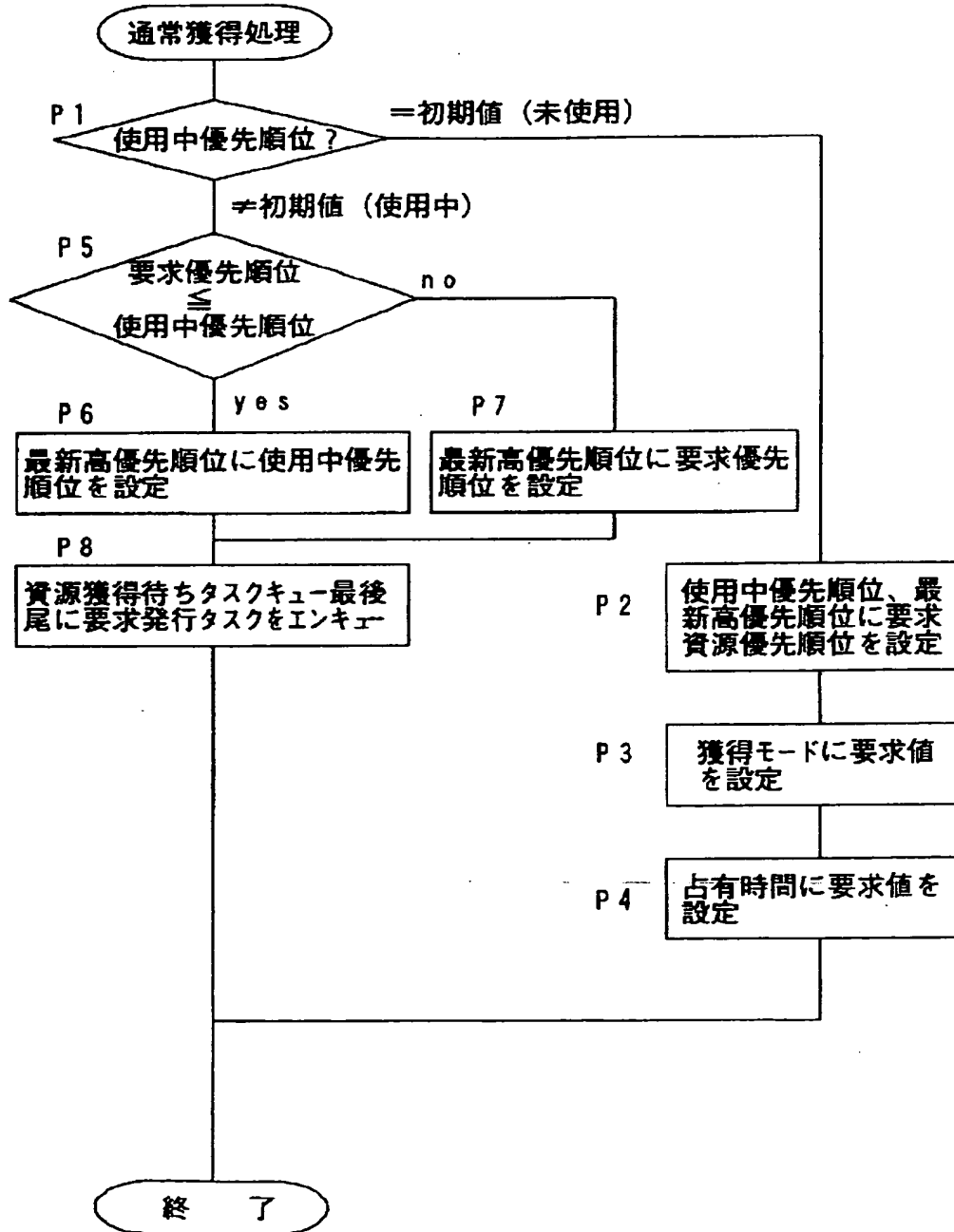
※ タスク B が資源を獲得

資源名	使用中優先順位	最新高優先順位	資源獲得モード	資源占有時間	資源獲得周期時間	資源獲得優先順位	資源獲得待ちタスクキュー	周期獲得待ちタスクキュー
						1	RP1	無し
R1	3	3	周期	20	50	2	RP2	無し
						3	RP3	タスク D
						:		
						n	RPn	無し

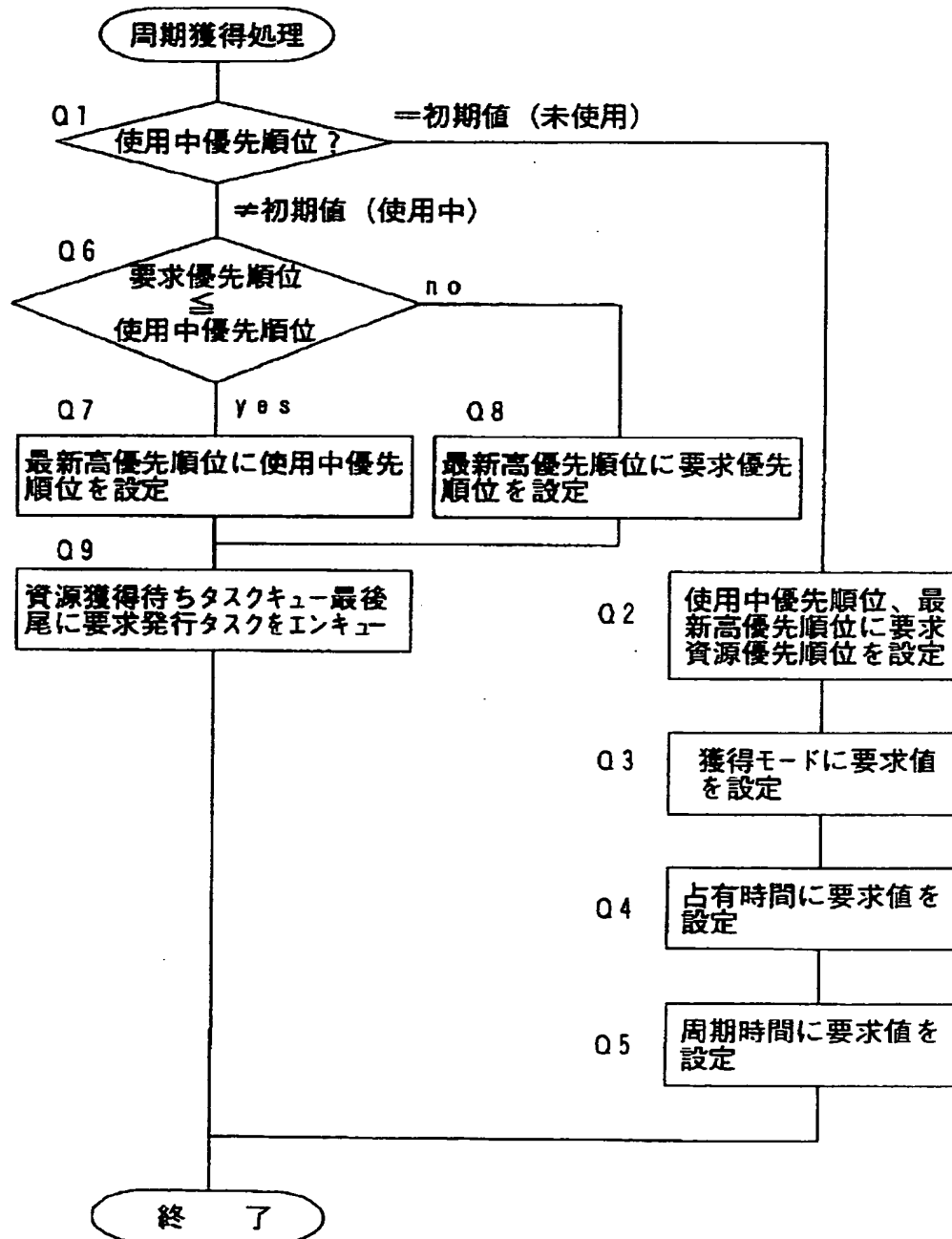
【図 5】



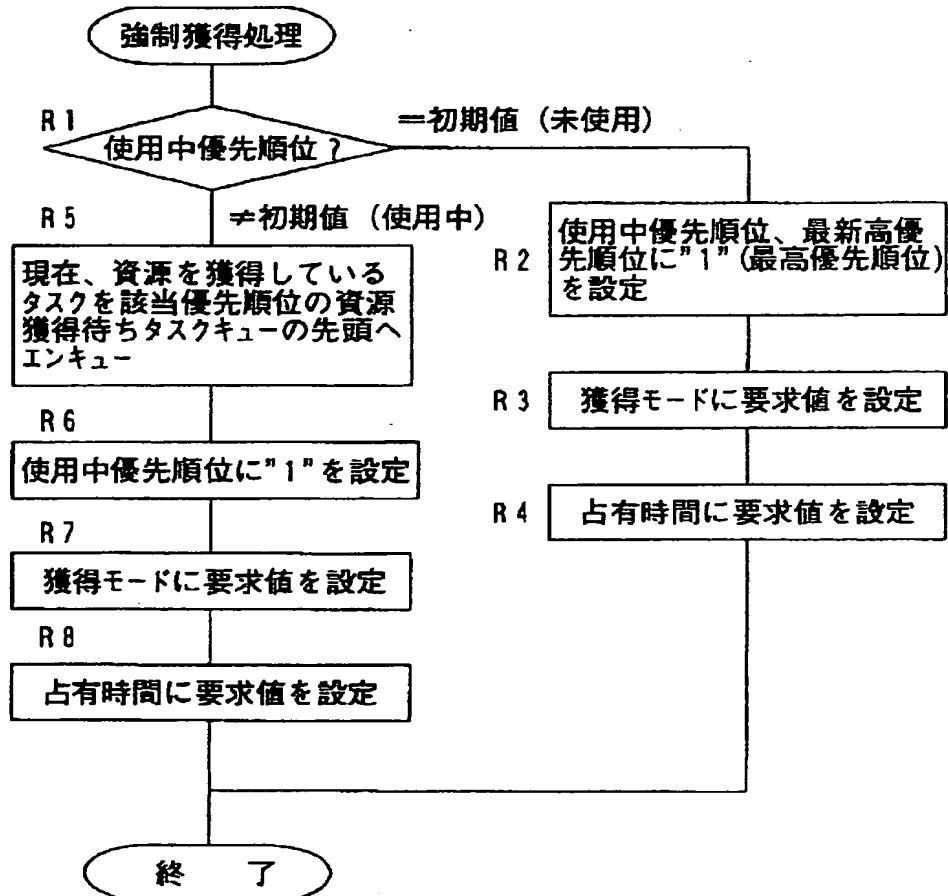
【図 6】



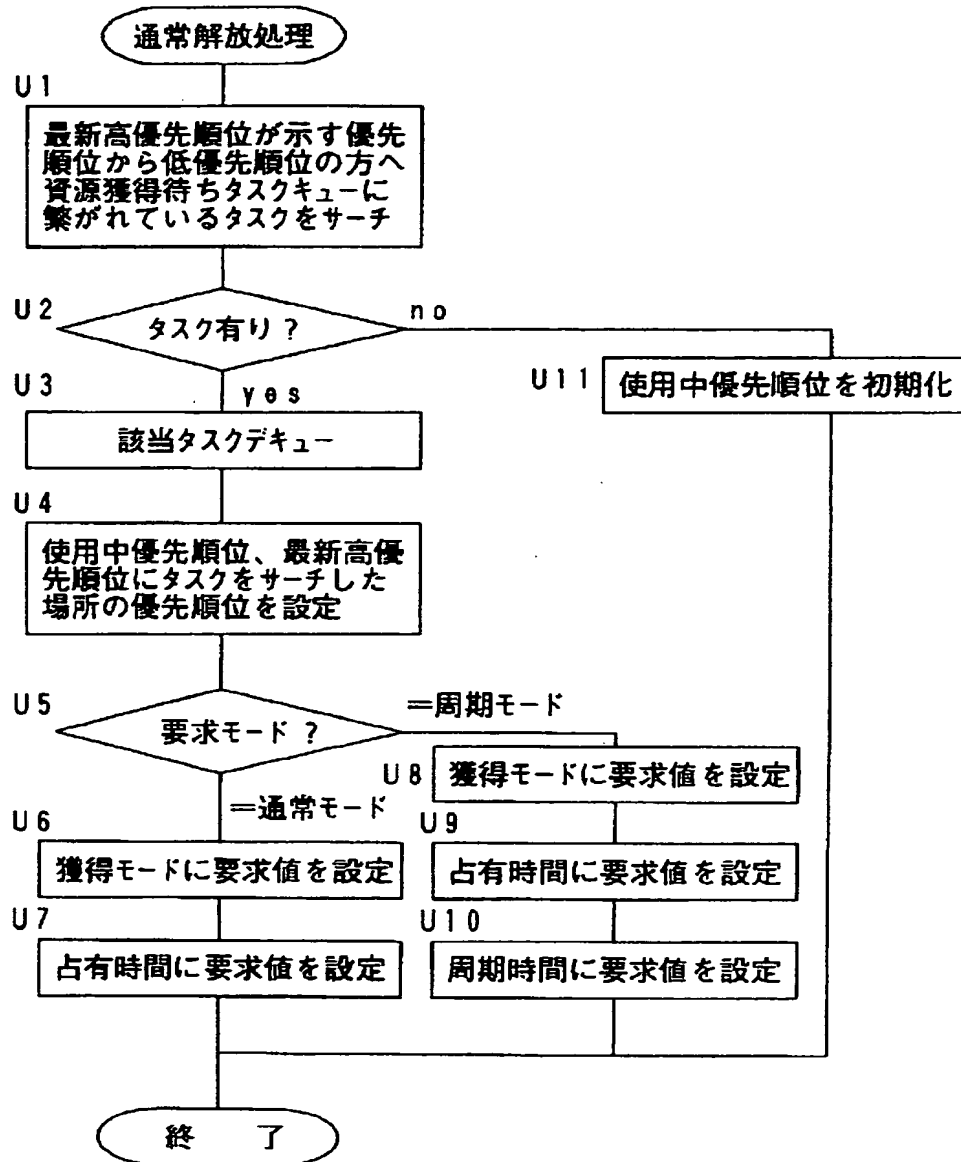
【図 7】



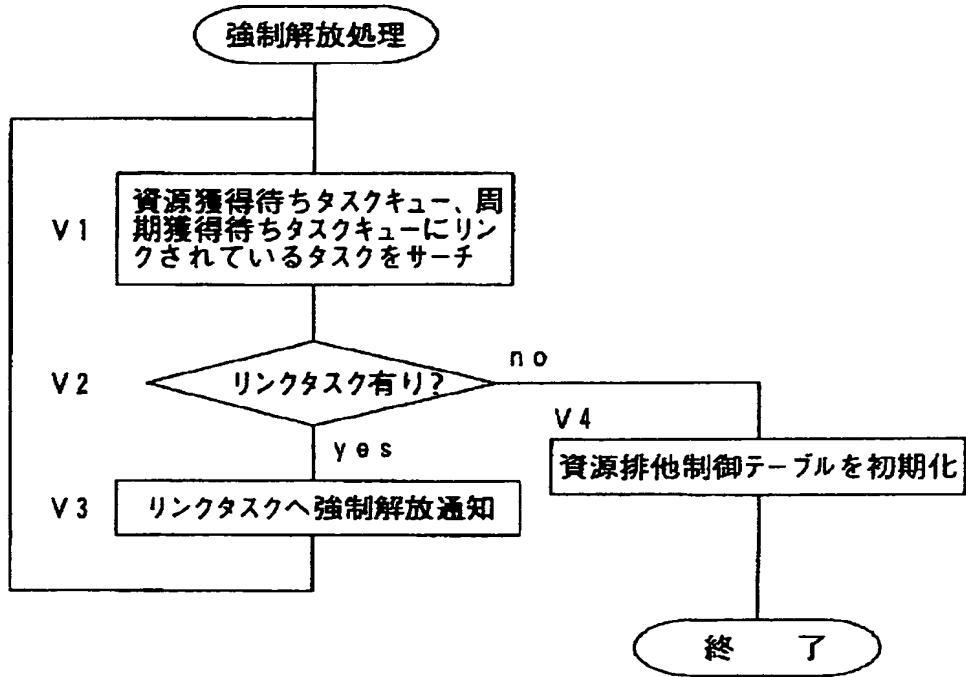
【図 8】



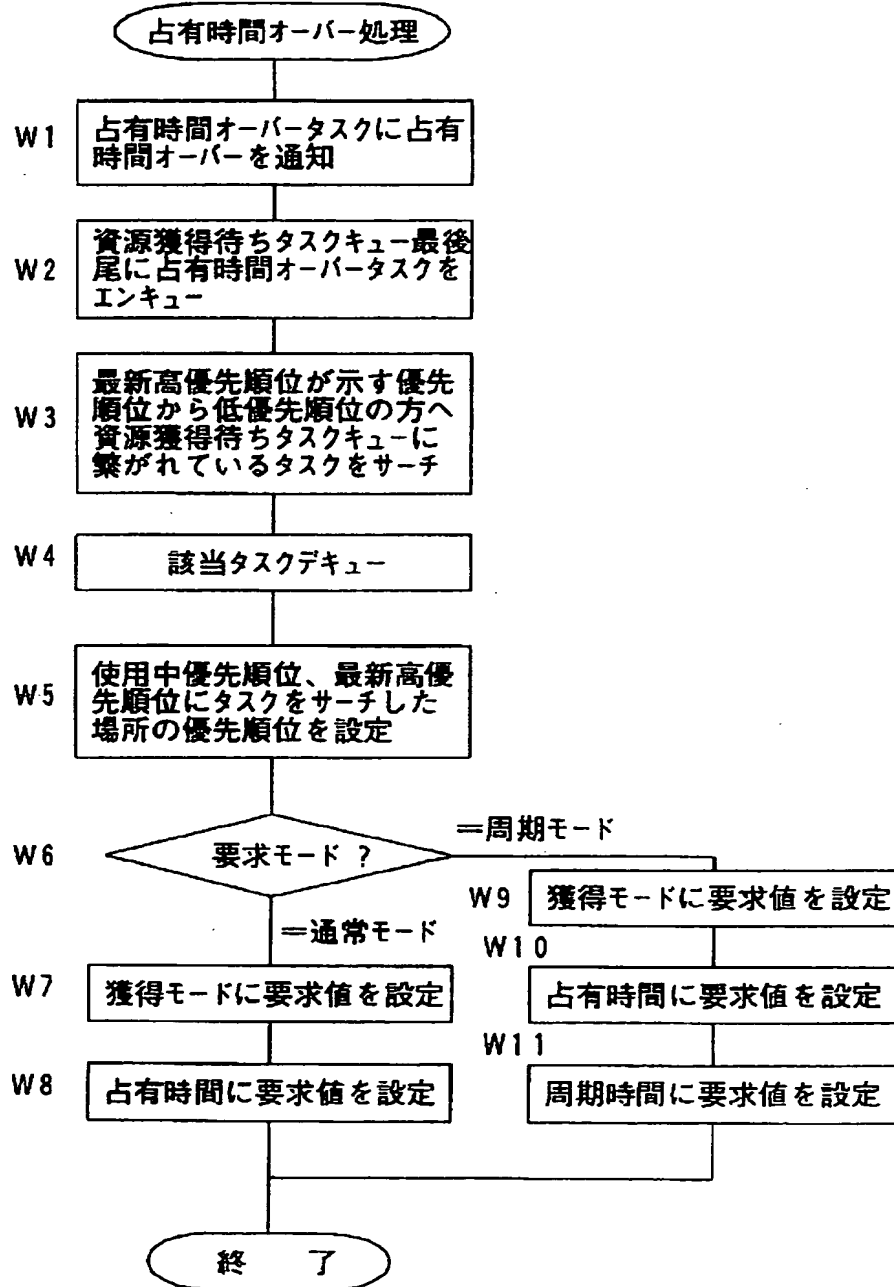
【図9】



【図 10】



【図 11】



【図12】

